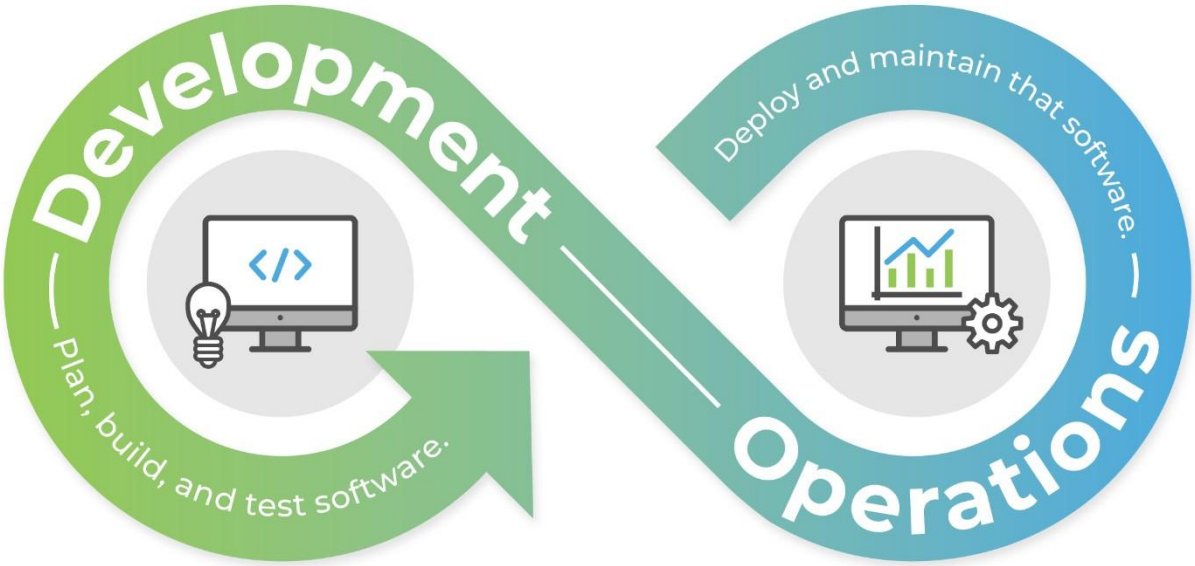


Introduction: What is DevOps?

Technical Definition:

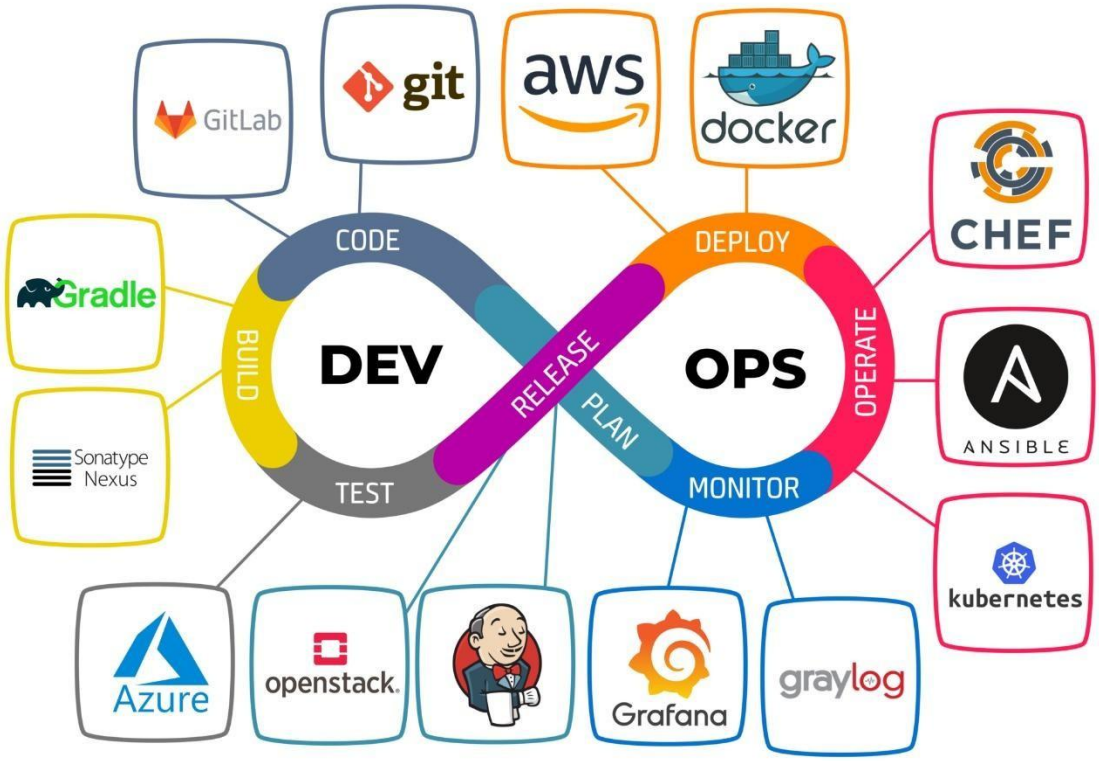
DevOps is a set of practices that combines software development (Dev) and IT operations (Ops). It aims to shorten the system development life cycle and provide continuous delivery with high software quality.

Down-to-Earth Explanation:

Think of DevOps like being a super-organized wedding planner (context: someone coordinating everything to go smoothly). Developers are like the designers who create the dream wedding outfit, and Ops are the decorators who make the venue functional. DevOps makes sure the dress fits, the cake arrives, and the music plays at the right time — all together!

Real-World Example:

Netflix uses DevOps practices to update features like recommendations or video quality smoothly without bothering users.



1. Foundations: Understand the Playground

1.1 What is DevOps?

Technical Definition:

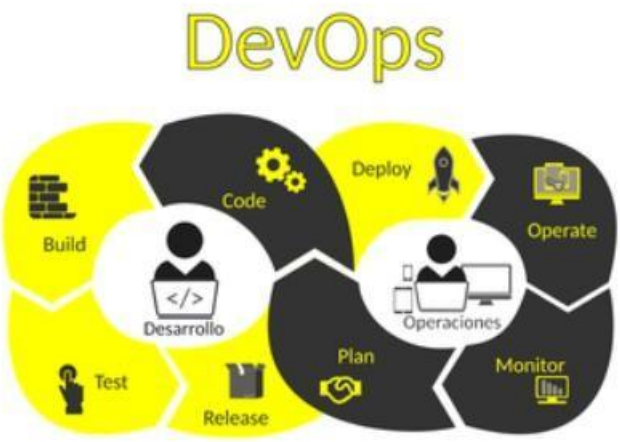
DevOps is a cultural and professional movement that emphasizes collaboration between software developers and IT operations professionals to deliver software quickly and reliably.

Down-to-Earth Explanation:

It's like running a kitchen with two main teams — chefs (developers) and waiters (operations). The DevOps role is like the restaurant manager (context: the one who makes sure both the food is cooked well and served hot). Everyone works together, no shouting, and the customers (users) are always happy.

Real-World Example:

Amazon pushes hundreds of small updates daily without downtime — DevOps ensures it's smooth and safe.



1.2 Computer Basics

Technical Definition:

Understanding basic computer operations like how operating systems work, what file systems do, and how networking connects computers is foundational for DevOps.

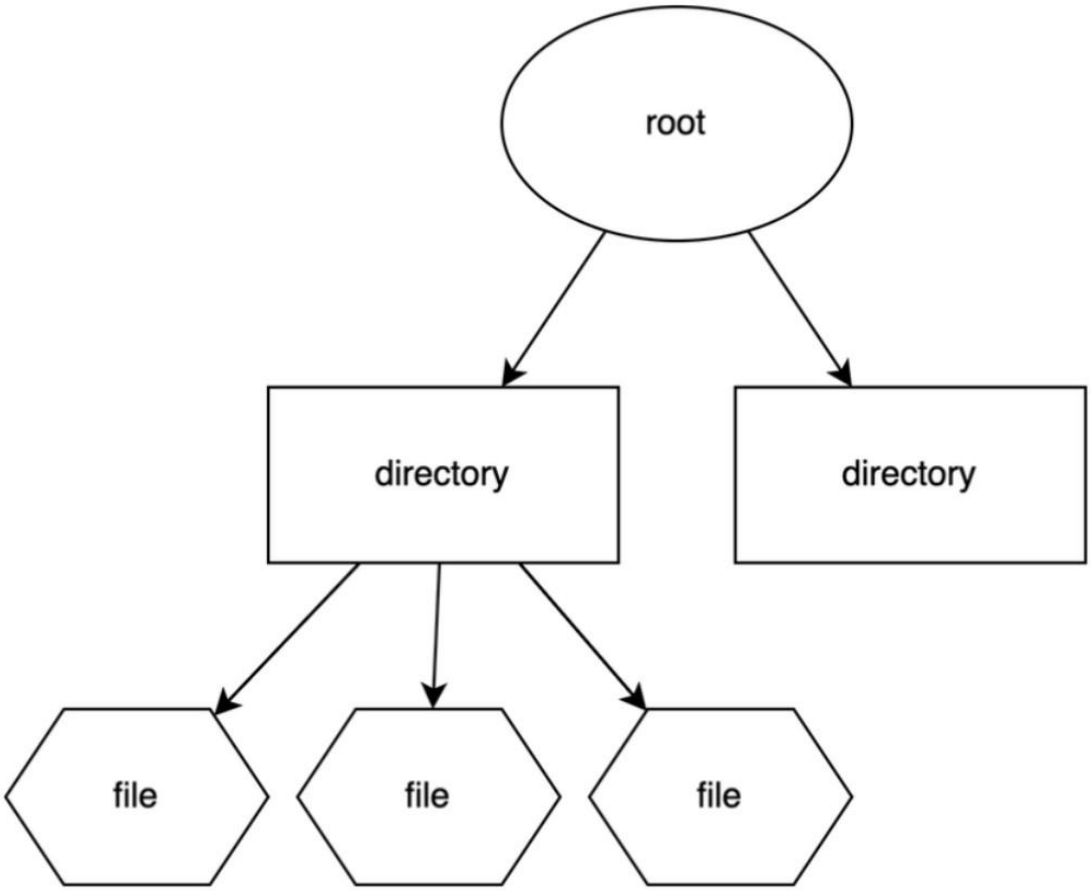
Down-to-Earth Explanation:

Think of your laptop like your home kitchen (context: where you prepare everything). Knowing how to switch on the stove (OS), organize your spices (file system), and connect to Wi-Fi (networking) is the starting point before you can host a dinner party (run apps or tools).

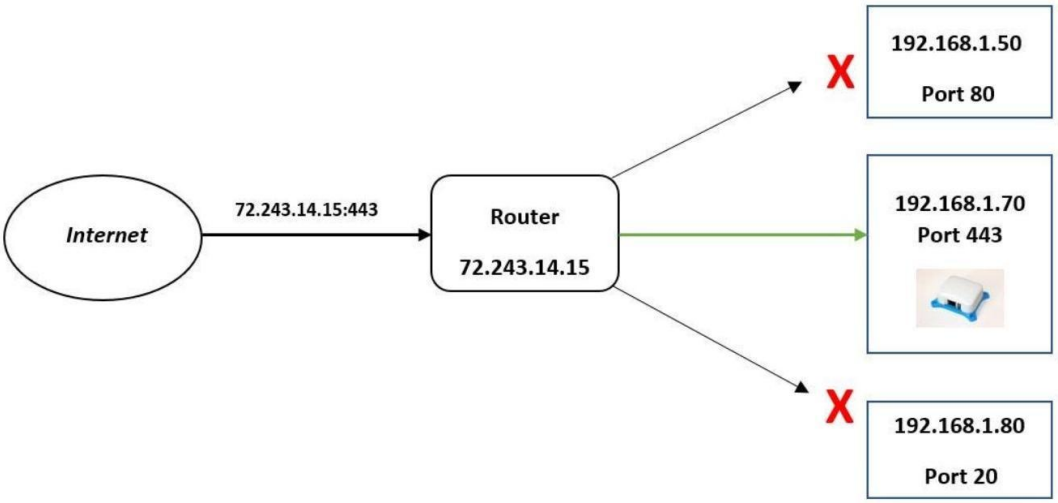
Real-World Example:

When setting up Zoom, adjusting camera/microphone permissions — that's already using OS-level knowledge!

OS Basic Services



File System Basic



Networking Basic

2. System Basics

2.1 Linux

Technical Definition:

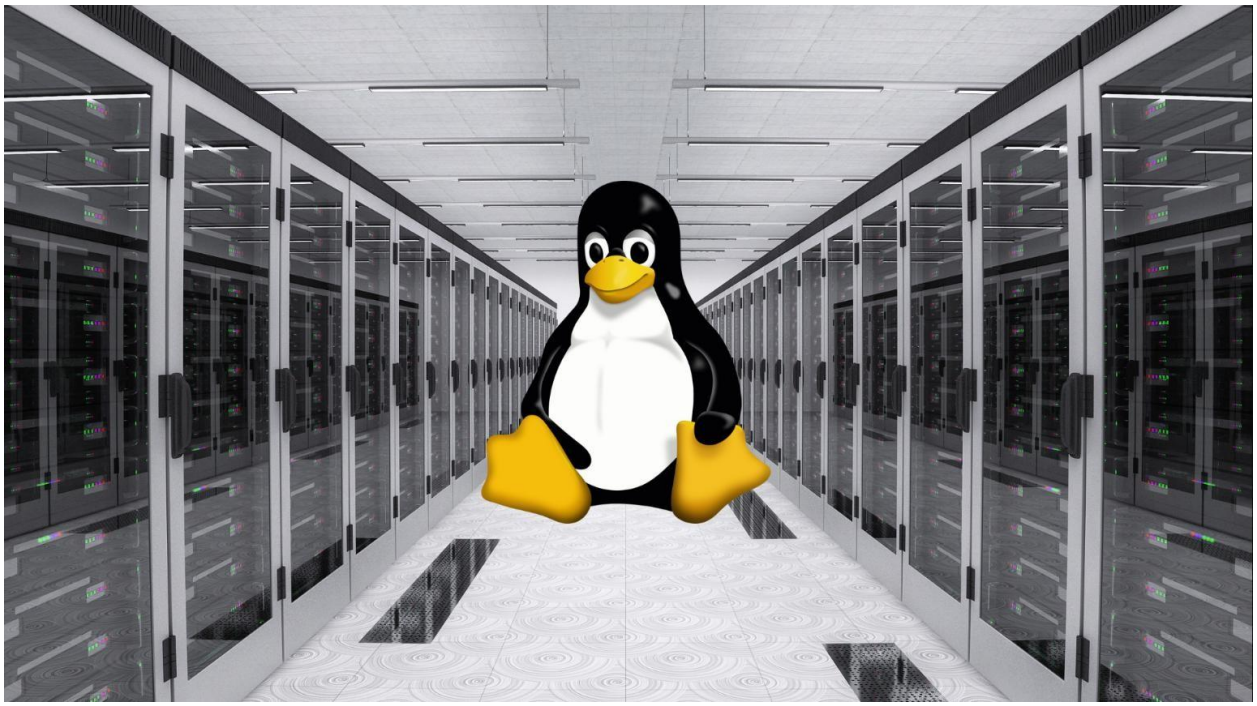
Linux is an open-source Unix-like operating system, popular in server and cloud environments because of its stability, security, and flexibility.

Down-to-Earth Explanation:

Linux is like a minimalist but powerful kitchen (context: fewer buttons, more power). Unlike Windows which shows everything with clicks and icons, Linux quietly does the job through commands — perfect for managing large-scale services (like websites and apps).

Real-World Example:

Websites like Facebook and Spotify run on Linux-based servers to handle millions of users.



2.2 Command Line

Technical Definition:

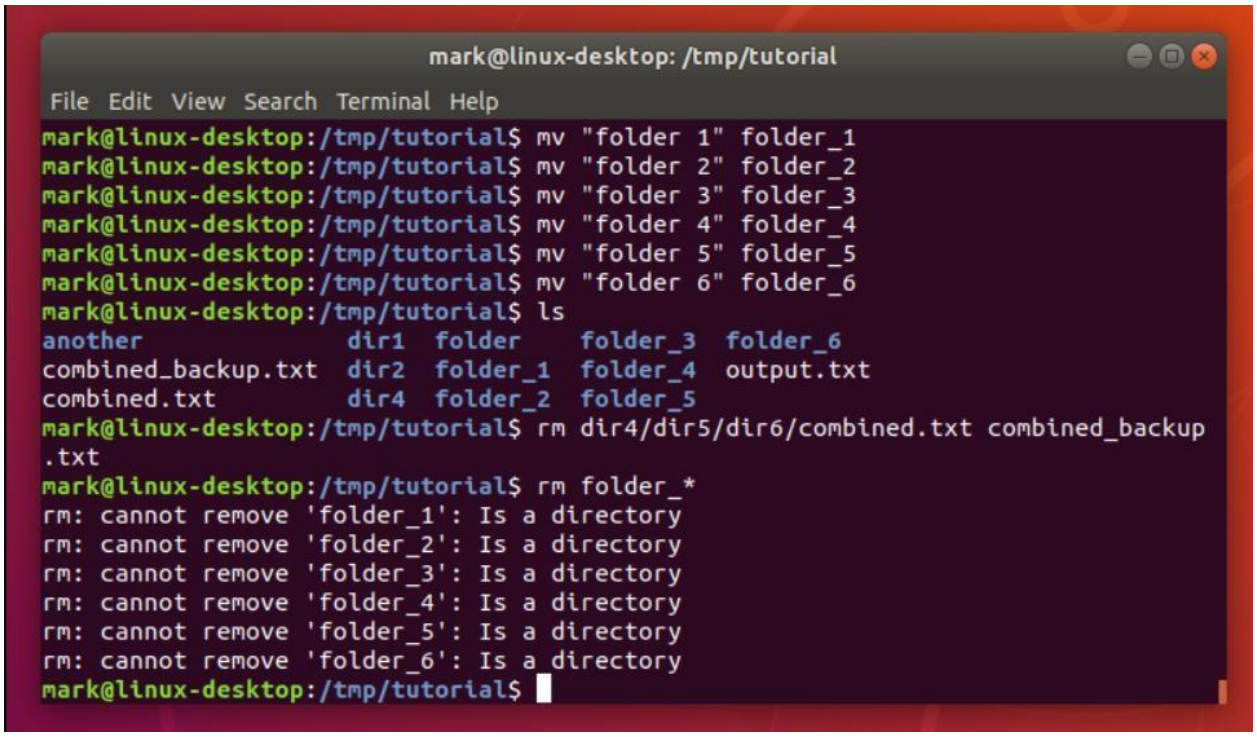
The Command Line Interface (CLI) allows users to type instructions directly to the computer, bypassing the graphical interface for faster, more controlled operations.

Down-to-Earth Explanation:

It's like sending voice commands to your smart assistant (context: "turn on the lights," becomes ls to list files). The computer obeys your typed commands instead of clicking around with a mouse — and it's faster once you get the hang of it!

Real-World Example:

Typing mkdir new-folder in the terminal is like asking your phone to "create a new photo album" — but more direct and flexible.


mark@linux-desktop: /tmp/tutorial
File Edit View Search Terminal Help
mark@linux-desktop:/tmp/tutorial$ mv "folder 1" folder_1
mark@linux-desktop:/tmp/tutorial$ mv "folder 2" folder_2
mark@linux-desktop:/tmp/tutorial$ mv "folder 3" folder_3
mark@linux-desktop:/tmp/tutorial$ mv "folder 4" folder_4
mark@linux-desktop:/tmp/tutorial$ mv "folder 5" folder_5
mark@linux-desktop:/tmp/tutorial$ mv "folder 6" folder_6
mark@linux-desktop:/tmp/tutorial$ ls
another dir1 folder folder_3 folder_6
combined_backup.txt dir2 folder_1 folder_4 output.txt
combined.txt dir4 folder_2 folder_5
mark@linux-desktop:/tmp/tutorial$ rm dir4/dir5/dir6/combined.txt combined_backup
.txt
mark@linux-desktop:/tmp/tutorial$ rm folder_*
rm: cannot remove 'folder_1': Is a directory
rm: cannot remove 'folder_2': Is a directory
rm: cannot remove 'folder_3': Is a directory
rm: cannot remove 'folder_4': Is a directory
rm: cannot remove 'folder_5': Is a directory
rm: cannot remove 'folder_6': Is a directory
mark@linux-desktop:/tmp/tutorial$

3. Tools of the Trade

3.1 Git & GitHub

Technical Definition:

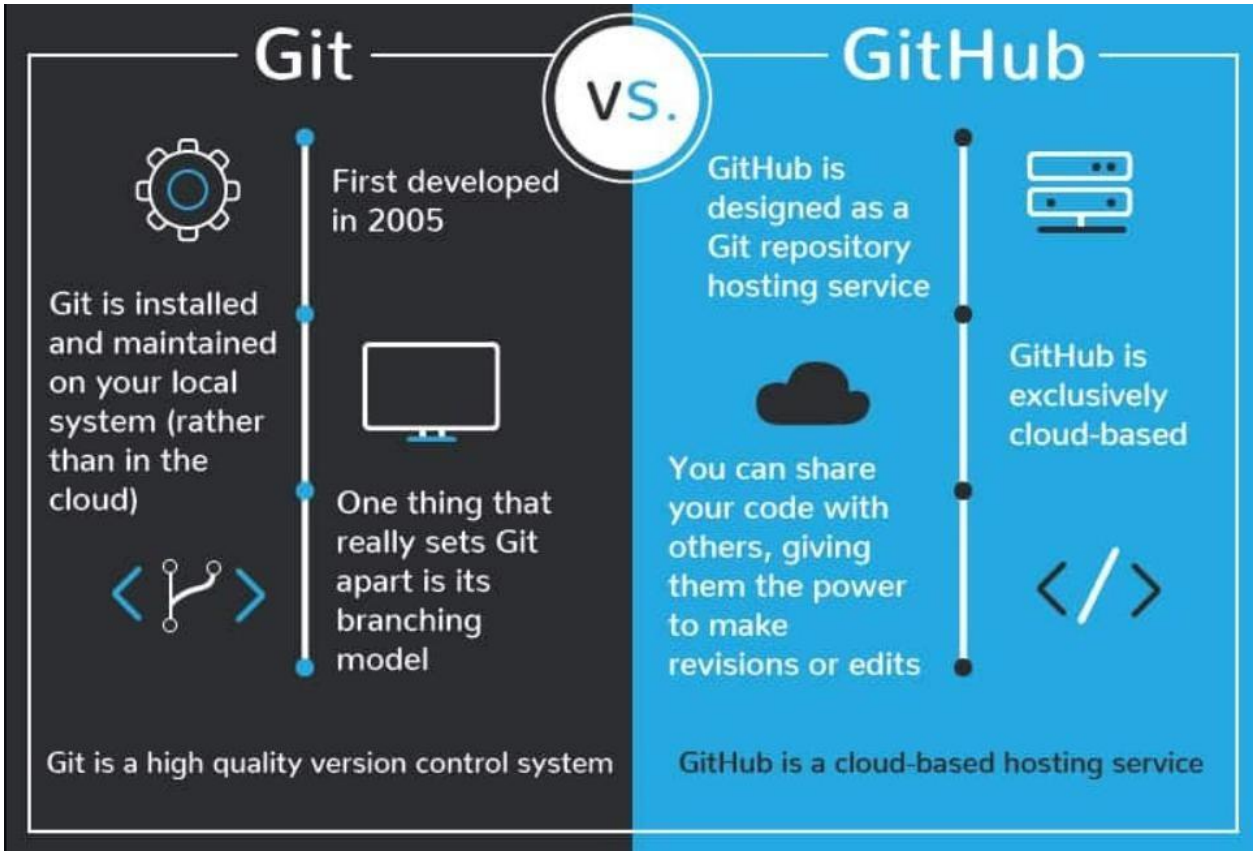
Git is a distributed version control system to track changes in source code. GitHub is a cloud-based platform to host and collaborate on Git repositories.

Down-to-Earth Explanation:

Imagine writing a novel and saving every version in a journal (Git). Now imagine sharing it with friends online where they can comment, edit, and contribute (GitHub) — it's like a shared diary with backup copies!

Real-World Example:

Teams at Canva work on the same codebase and collaborate through GitHub without overwriting each other's work.


Git
VS.
GitHub
First developed in 2005
Git is installed and maintained on your local system (rather than in the cloud)
One thing that really sets Git apart is its branching model
GitHub is designed as a Git repository hosting service
GitHub is exclusively cloud-based
You can share your code with others, giving them the power to make revisions or edits
Git is a high quality version control system
GitHub is a cloud-based hosting service

3.2 Shell Scripting

Technical Definition:

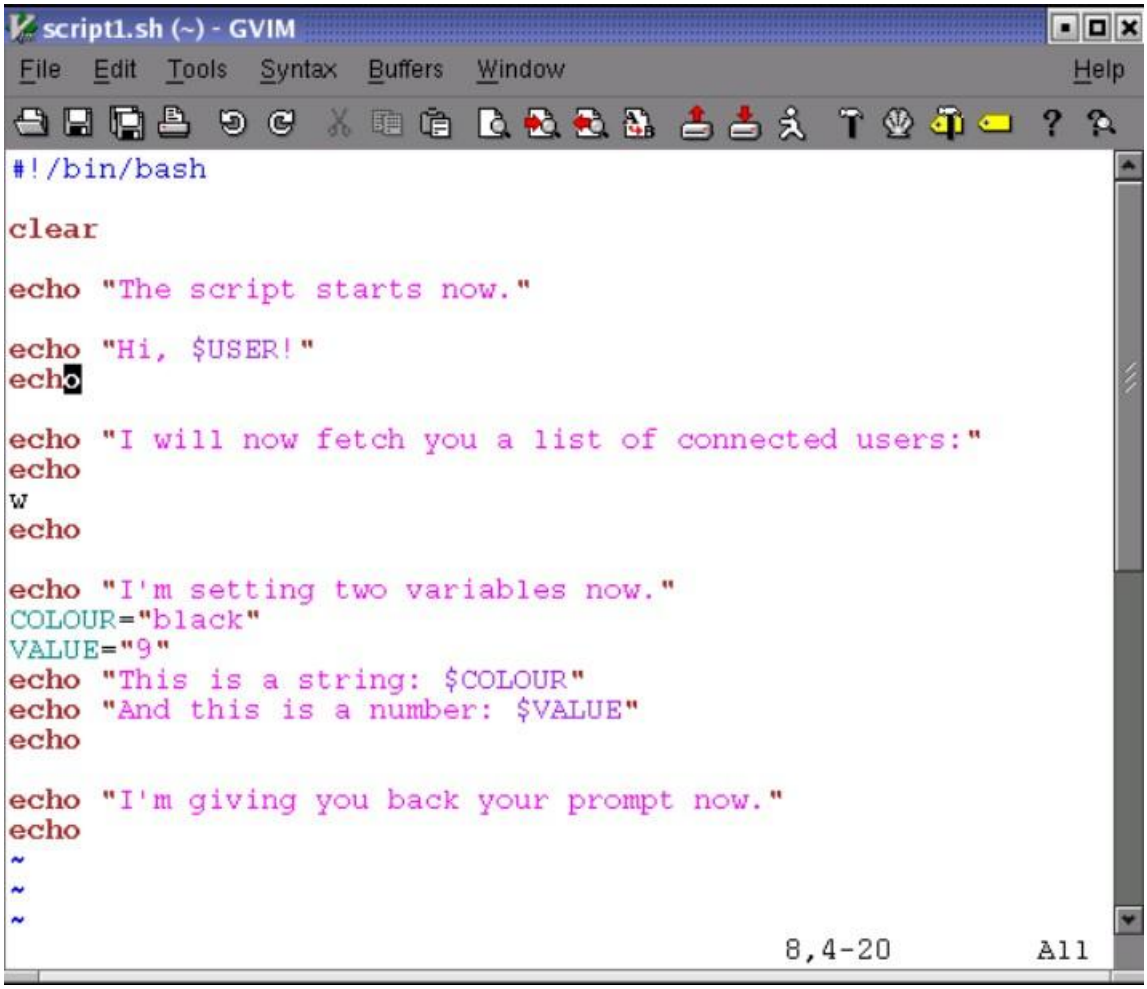
Shell scripting involves writing command-line instructions in a file to automate tasks in Unix/Linux environments.

Down-to-Earth Explanation:

Think of it like setting a morning routine alarm (context: you don't want to brush, make coffee, and feed the cat manually every day). Instead, you set a script (checklist) that says:
"Do all this — every morning — without asking me."

Real-World Example:

A company uses a shell script to automatically clean up temporary files every night at midnight.



4. Continuous Integration and Continuous Deployment (CI/CD)

4.1 Continuous Integration (CI)

Technical Definition:

CI is a practice where developers regularly merge their code changes into a shared repository, followed by automated builds and tests to detect issues early.

Real-World Vibes (Imagine this):

Think of CI like a group project where everyone adds their part to the same essay (code). Instead of waiting till the last minute to check if it all fits, you keep reviewing each small change instantly — like your best friend reading your paragraph right after you write it, so no embarrassing typos go unnoticed! (context: catching errors early helps avoid bigger problems later)

Real-World Example:

Google's developers push code frequently, and automated tests run immediately to ensure new features don't break anything.

4.2 Continuous Deployment (CD)

Technical Definition:

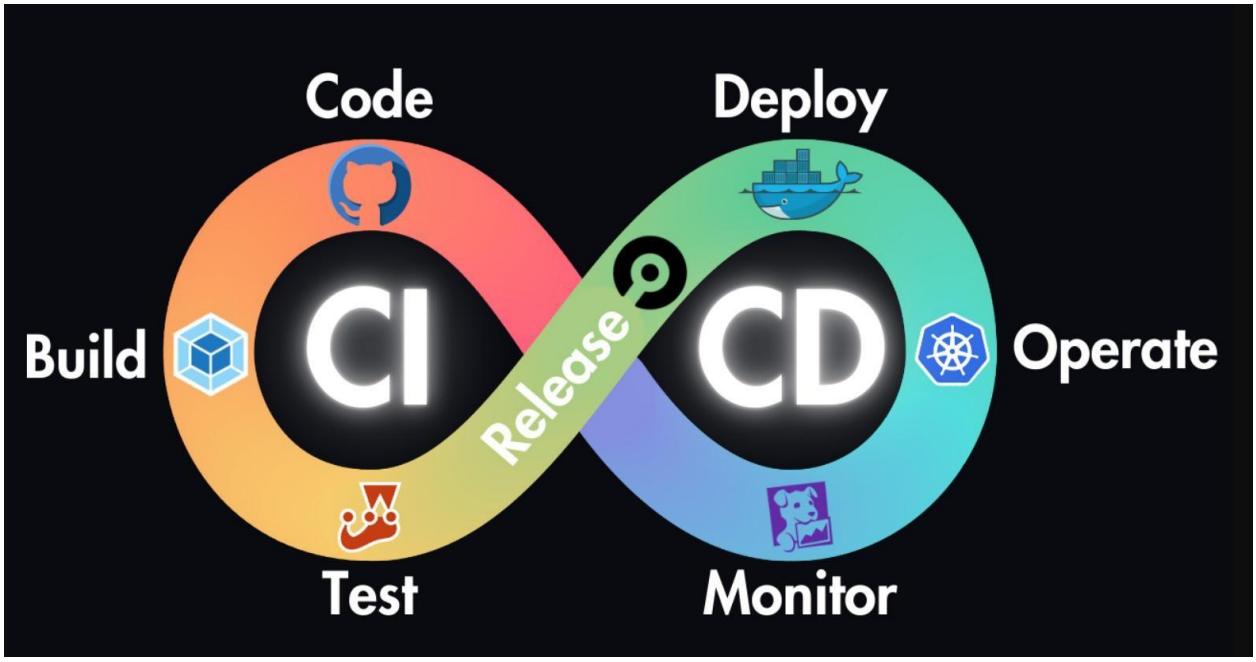
CD is the practice of automatically deploying code changes to production after passing all tests, enabling rapid and reliable software releases.

Real-World Vibes:

Imagine your Instagram feed updating automatically the moment your friend posts a picture — without you refreshing manually (context: software updates happening seamlessly and immediately). CD means the app gets better instantly and users see new features without waiting.

Real-World Example:

Netflix deploys new features and fixes daily without downtime — viewers barely notice anything changed!



5. Cloud Computing Basics

5.1 What is Cloud?

Technical Definition:

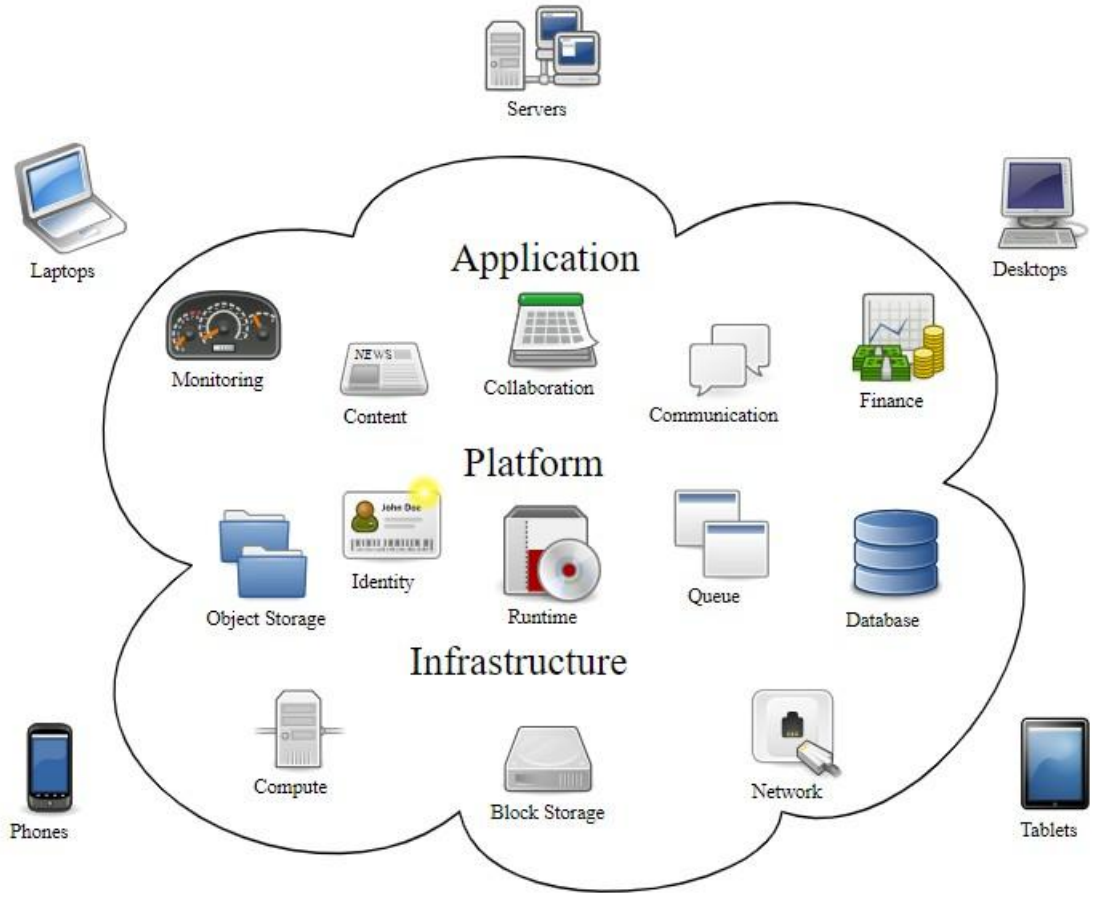
Cloud computing provides on-demand computing resources (servers, storage, databases) over the internet instead of using local hardware.

Real-World Vibes:

Think of cloud like renting a fancy wardrobe instead of buying your own clothes (context: instead of owning a big expensive computer, you borrow power from someone else's computer through the internet). You only pay for what you use and can switch styles anytime!

Real-World Example:

Spotify stores your music and playlists on the cloud so you can listen anywhere, anytime — no big hard drive needed on your phone.



5.2 Popular Cloud Providers

Technical Definition:

Top cloud service providers include Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), offering various computing and storage services.

Real-World Vibes:

Think of AWS, Azure, and GCP as three giant malls where you can shop for any tech tool you need — from virtual machines (computers) to databases (organized shelves for your stuff). Each mall has its unique brands and perks!

Real-World Example:

Netflix uses AWS to handle its huge video streaming service worldwide.

Cloud Services	aws		
Virtual Servers	Instances	VM Instances	VMs
Platform-as-a-Service	Elastic Beanstalk	App Engine	Cloud Services
Serverless Computing	Lambda	Cloud Functions	Azure Functions
Docker Management	ECS	Container Engine	Container Service
Kubernetes Management	EKS	Kubernetes Engine	Kubernetes Service
Object Storage	S3	Cloud Storage	Block Blob
Archive Storage	Glacier	Coldline	Archive Storage
File Storage	EFS	ZFS / Avere	Azure Files
Global Content Delivery	CloudFront	Cloud CDN	Delivery Network
Canaged Data Warehouse	Redshift	Big Query	SQL Warehouse

6. Containerization with Docker

6.1 What is Docker?

Technical Definition:

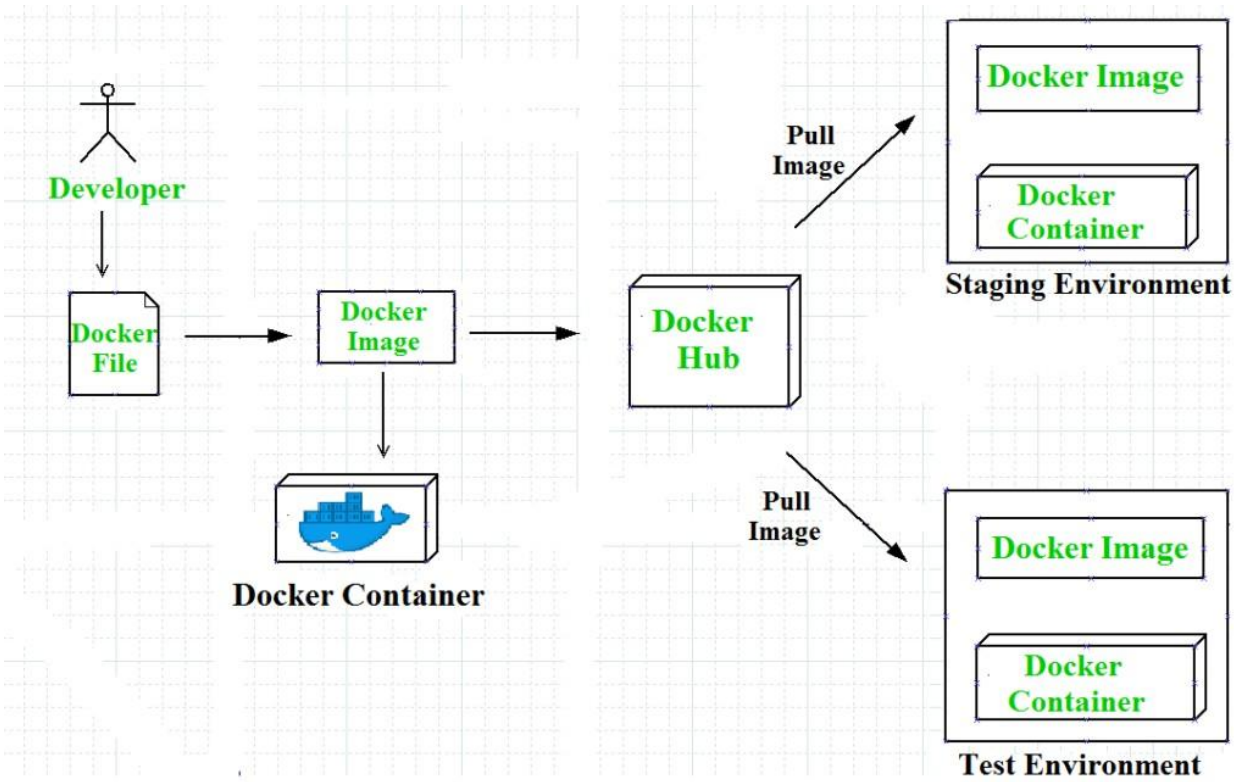
Docker is a platform that uses containers to package applications with all their dependencies to run reliably across different computing environments.

Real-World Vibes:

Imagine a magical lunchbox that keeps your food (app) fresh and perfectly cooked, no matter which kitchen (computer) you take it to. Docker wraps everything the app needs into one container so it never breaks because of missing ingredients!

Real-World Example:

Developers use Docker to make sure apps run the same on their laptop, testing server, and production environment.



6.2 Containers vs Virtual Machines (VMs)

Technical Definition:

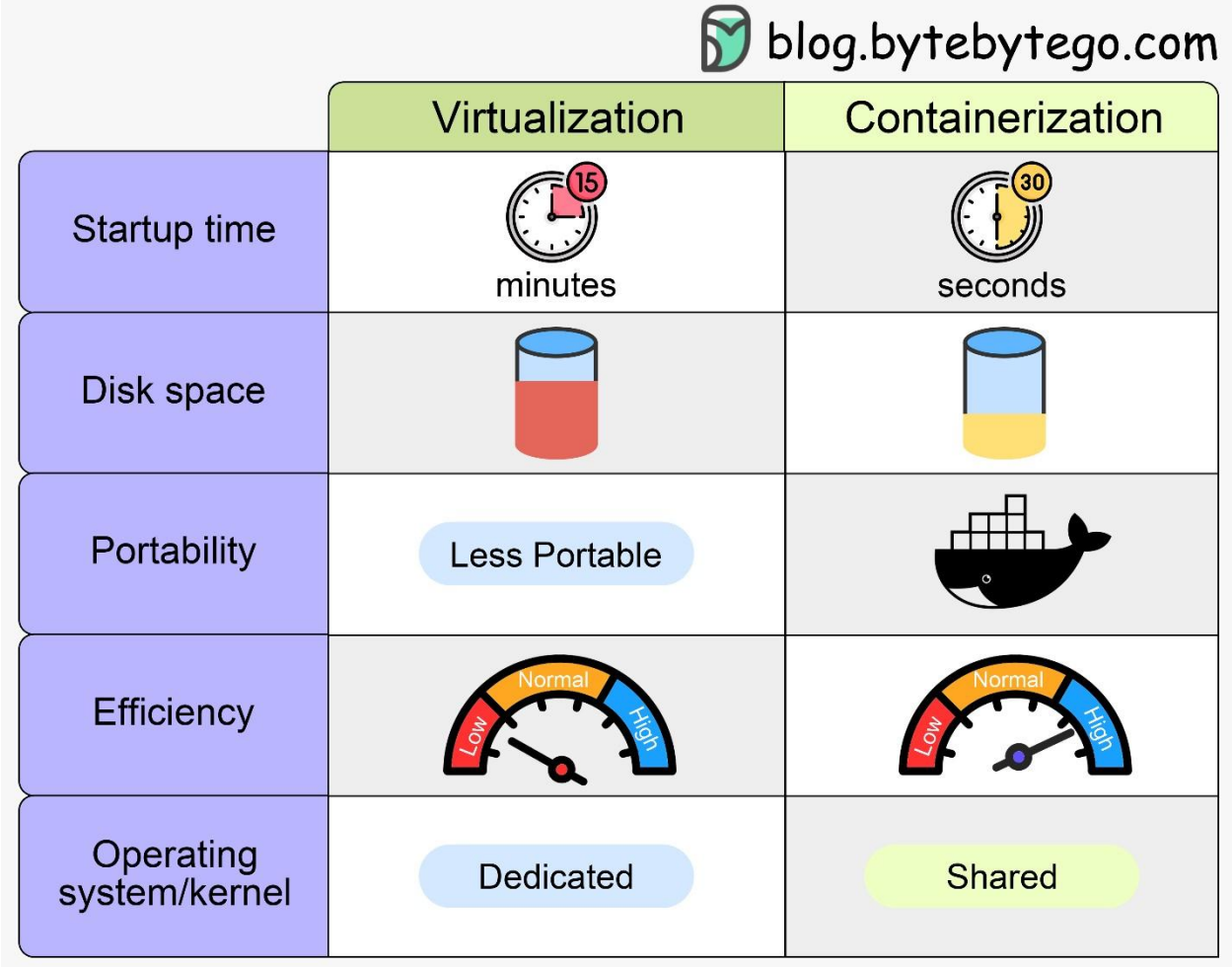
Containers share the host OS kernel and isolate apps at the process level, while VMs run a full guest OS with hardware virtualization.

Real-World Vibes:

Containers are like neat bento boxes sharing the same fridge (OS), while VMs are like separate mini fridges , each with its own freezer and fridge inside. Containers start faster and use less space.

Real-World Example:

A company running hundreds of containers can save a lot on resources compared to spinning up hundreds of VMs.



7. Orchestration with Kubernetes

7.1 What is Kubernetes?

Technical Definition:

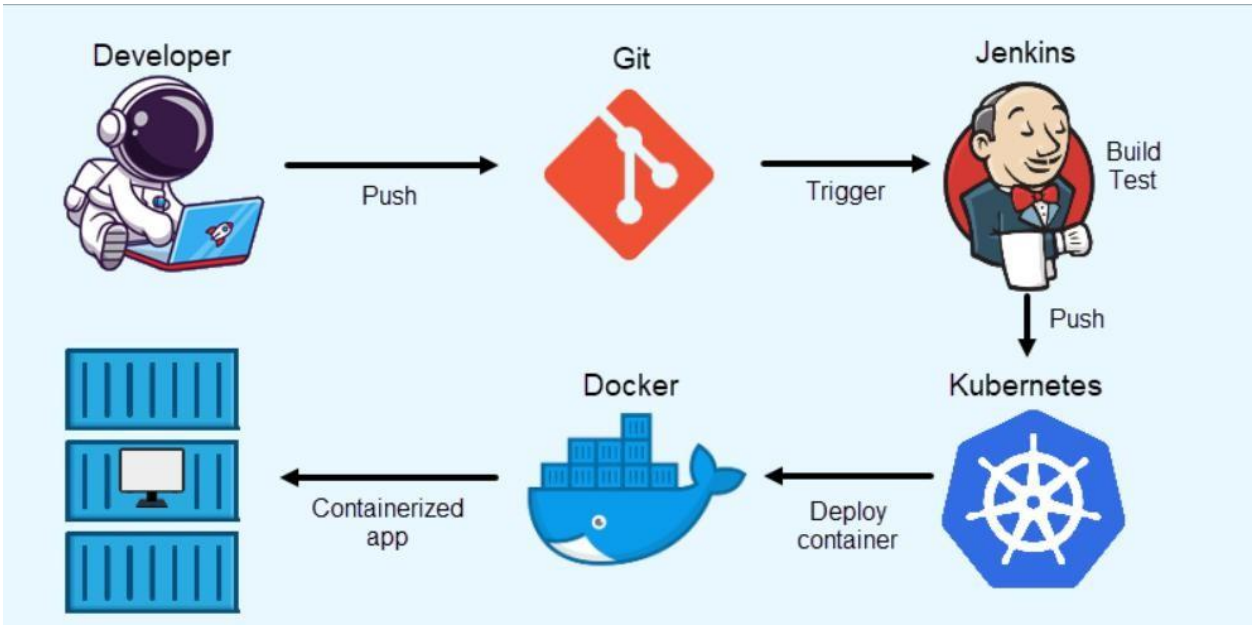
Kubernetes is an open-source platform to automate deploying, scaling, and managing containerized applications.

Real-World Vibes:

Imagine a super-smart party planner (Kubernetes) managing hundreds of food delivery boxes (containers). It decides who gets what, when, and where — ensuring no food runs out and every guest is happy (apps run smoothly and scale with demand).

Real-World Example:

Companies like Airbnb use Kubernetes to manage their complex web services across many servers effortlessly.



7.2 Pods, Nodes, Clusters

Technical Definition:

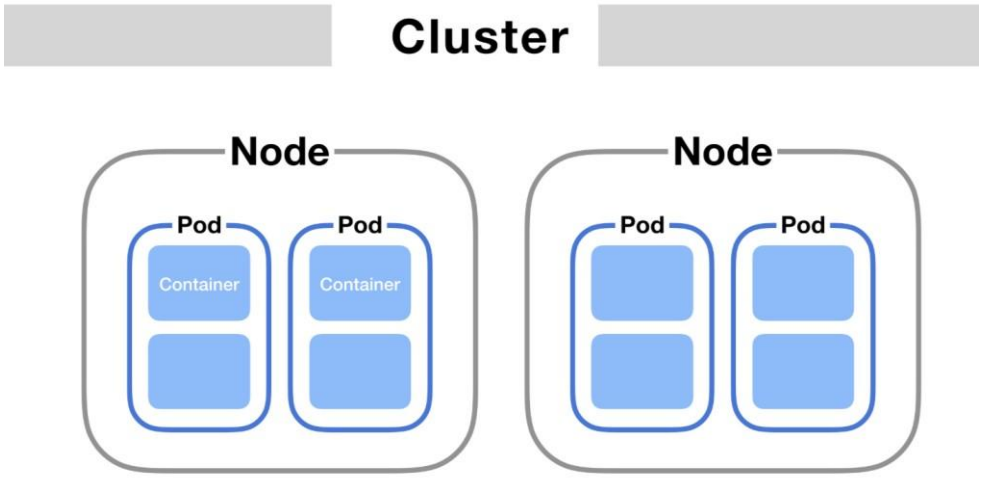
- Pods are the smallest deployable units in Kubernetes, hosting one or more containers.
- Nodes are individual worker machines (servers). • Clusters are groups of nodes managed together.

Real-World Vibes:

Think of pods as lunchboxes , nodes as delivery vans , and clusters as the whole fleet of vans working together to deliver lunch to everyone on time.

Real-World Example:

When traffic spikes, Kubernetes adds more vans (nodes) and lunchboxes (pods) to serve hungry users.



8. Monitoring and Logging

8.1 Why Monitor?

Technical Definition:

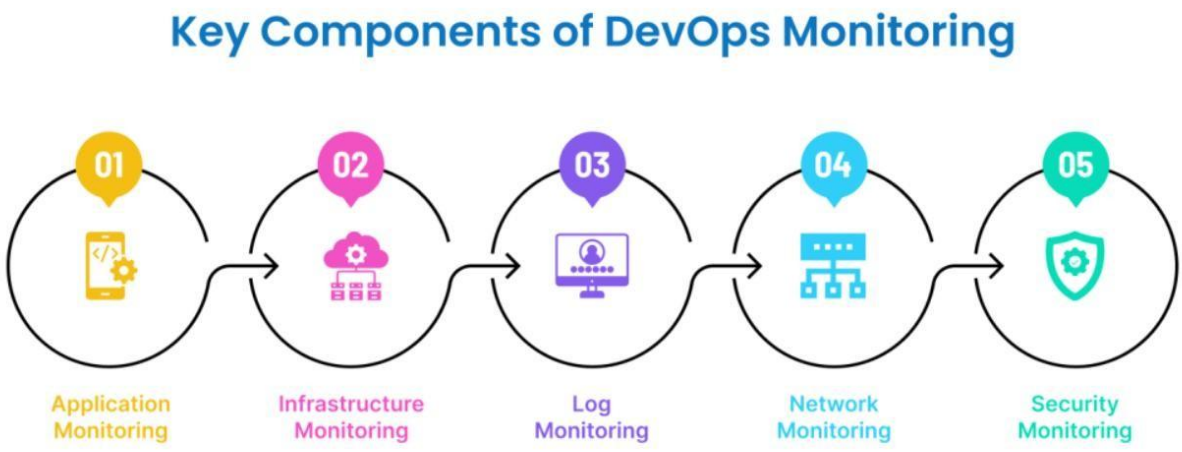
Monitoring tracks the health, performance, and availability of applications and infrastructure to detect issues early.

Real-World Vibes:

It's like having a fitness tracker on your car — telling you when oil is low or the tire pressure is bad before it breaks down on the highway. Monitoring keeps your apps healthy and speedy.

Real-World Example:

Netflix monitors streaming quality and server health to fix problems before viewers notice.



8.2 Other Tools

Technical Definition:

Tools like Prometheus (metrics), Grafana (dashboards), and ELK Stack (Elasticsearch, Logstash, Kibana) are widely used for monitoring and logging.

Real-World Vibes:

Prometheus is like the heart rate monitor , Grafana is your personal health dashboard , and ELK is the detective gathering clues from logs to find what went wrong.

Real-World Example:

A fintech app uses these tools to ensure transactions happen smoothly and track errors instantly.

Created by

Chetan Raj

For more such docs, click on the link or scan the barcode
https://archive.org/details/@chetan_raj869